In the Claims

Applicant has submitted a new complete claim set showing amended claims with insertions indicated by underlining and deletions indicated by strikeouts and/or double bracketing.

Please cancel claim 15 without prejudice or disclaimer.

Please amend pending claims 1, 2, 8, 9, 16-21, 22, 27 and 28 as noted below.

CLAIMS

1. (Amended) A method for processing a signal value in a digital signal processor, the method comprising [[the step of]]:

in response to a single instruction that specifies at least a signal value and a despreading code[[,]] decoding the signal value by:

[[multiplying]] performing a multiplication of bits of the signal value by bits of the despreading code,

performing an addition summing the results of the multiplication, and

storing the results of the addition.

2. (Amended) A method as defined in claim 1 further comprising, in response to the single instruction, the step of adding the result of the [multiplying] decoding to a result from a previous [multiplication] decoding.

3. (Original) A method as defined in claim 1, wherein the despreading code has a spreading factor divisible by 4.

4. (Original) A method as defined in claim 1, wherein the despreading code is divided into code segments, each code segment having comprising a 2 bit complex code comprising 1 real bit and 1 imaginary bit.

5. (Original) A method as defined in claim 4, wherein a set code bit represents a value of –1 and a clear code bit represents a value of +1.

6. (Original) A method as defined in claim 1, wherein the signal value comprises 16 bits.

7. (Original) A method as defined in claim 6, wherein the signal value comprises 8 real bits and 8 imaginary bits.

8. (Amended) A method for calculating a data set in a digital signal processor, the method comprising the steps of:

in response to [[one or more]] a single instruction[[s]] that [[specify]] specifies at least a signal value and a set of codes:

for each one of the set of codes multiplying the signal value by one of the set of codes;

summing results of the multiplying; and

producing a data set resulting from the summing.

9. (Amended) A method as defined in claim 8, wherein, in response to the single instruction, the summing comprises summing results of the multiplying with the results of a multiplying by a previous set of codes.

10. (Original) A method as defined in claim 8, wherein the set of codes has a spreading factor divisible by 4.

11. (Original) A method as defined in claim 8, wherein each one of the set of codes is a 2 bit complex code comprising 1 real bit and 1 imaginary bit.

12. (Original) A method as defined in claim 11, wherein a set code bit represents a value of –1 and a clear code bit represents a value of +1.

13. (Original) A method as defined in claim 8, wherein the signal value comprises 16 bits.

14. (Original) A method as defined in claim 13, wherein the signal value comprises 8 real bits and 8 imaginary bits.

15. (Cancelled).

16. (Amended) A digital signal processor [[as defined in claim 15, wherein, in response to an execution of the decoding instruction the digital signal processor]] comprising:

a memory for storing instructions and operands for digital signal computations;

a program sequencer for generating instruction addresses for fetching selected ones of said instructions from said memory;

a computation block comprising a register file for temporary storage of operands and results and an execution block for executing a single decoding instruction that specifies a data signal and a code, said execution block comprising a complex multiply and accumulate engine for multiplying portions of the data signal by the code and accumulating the results; and

wherein, in response to an execution of the single decoding instruction the digital signal processor decodes the data signal by:

[[performs a set of complex multiplies on portions of the data signal and portions of the code; and

sums the results of the complex multiplies]]

performing a multiplication of bits of the data signal value by bits of the despreading code,

performing an addition summing the results of the multiplication, and

storing the results of the addition.

17. (Amended) [[A method]] A digital signal processor as defined in claim [[15]] 16, wherein the code has a spreading factor divisible by 4.

18. (Amended) [[A method]] A digital signal processor as defined in claim [[15]] 16, wherein the code is divided into code segments, each code segment having comprising a 2 bit complex code comprising 1 real bit and 1 imaginary bit.

19. (Amended) [[A method]] A digital signal processor as defined in claim 18, wherein a set code bit represents a value of –1 and a clear code bit represents a value of +1.

20. (Amended) [[A method]] A digital signal processor as defined in claim [[15]] 16, wherein the data signal comprises 16 bits.

21. (Amended) [[A method]] A digital signal processor as defined in claim [[15]] 16, wherein the data signal comprises 8 real bits and 8 imaginary bits.

22. (Amended) A method for calculating output data in a digital signal processor, the method comprising the steps of:

in response to [[one or more]] a single instruction[[s]] that [[specify]] specifies at least a set of complex first operands each one of the first operands comprising 8 real bits and 8 imaginary bits and a set of complex second operands each one of the second operands comprising 1 real bit and 1 imaginary bit:

for each one of the second operands performing a complex multiplication of one of the first operands by one of the second operands;

summing results of the multiplying over the set of second operands; and

producing as an output a set of data resulting from the summing.

23. (Original) A method as defined in claim 22, wherein a set bit in one of the second operands represents a value of –1 and a clear bit in one of the second operands represents a value of +1.

24. (Original) A method as defined in claim 22, wherein the set of complex second operands comprises a despreading code.

25. (Original) A method as defined in claim 22, wherein the set of complex first operands comprises an incoming data signal.

26. (Original) A method as defined in claim 25, wherein the incoming data signal is a voice transmission signal.

27. (Amended) A method for processing a signal value in a digital signal processor, comprising the step of:

in response to a single instruction specifying a complex signal value and a two bit complex code segment [[specified by an instruction]], performing a complex multiply of the signal value by the code segment to provide a processed data value.

28. (Amended) A method for processing signal values in a digital signal processor comprising the steps of:

(a) in response to a single instruction specifying a set of complex signal values and a corresponding set of complex code segments [[specified by an instruction]], performing a complex multiply of each signal value by a corresponding code segment to provide a set of intermediate values; and

(b) in further response to said single instruction performing complex addition of the intermediate values to provide a processed signal value.

29. (Original) A method as defined in claim 28 further comprising the steps of repeating steps (a) and (b) for a plurality of sets of complex signal values to provide a stream of processed signal values.

30. (Original) A method as defined in claim 28 further wherein each of the complex code segments is a two bit complex code.